REMARKS

Prior to this response, Claims 1-45 were pending in the application. By this amendment, Claims 1, 3, 24, 26, 32-37 and 39-43 are amended. No claims are added or cancelled. Hence, Claims 1-45 are currently pending in the application.

SUMMARY OF THE REJECTIONS/OBJECTIONS

Claims 3, 26, 32-37 and 39-43 were objected to because of formalities.

Claims 1-13, 16-36, and 39-45 were rejected under 35 U.S.C. §102(e) as allegedly anticipated by Chau et al. ("*Chau*"; U.S. Patent No. 6,721,727); and Claims 14, 15, 37 and 38 were rejected under 35 U.S.C. §103(a) as allegedly unpatentable over *Chau* in view of Ng et al. ("*Ng*"; U.S. Patent No. 6,385,618) and further in view of Bowen et al. ("*Bowen*"; U.S. Patent No. 6,094,694).

THE REJECTIONS BASED ON THE PRIOR ART

Rejection under 35 U.S.C. §102(e)

Claims 1-13, 16-36, and 39-45 were rejected under 35 U.S.C. §102(e) as allegedly anticipated by *Chau*. This rejection is traversed.

(A) Priority Date

As a preliminary matter, Applicants traverse the priority date given the present application. As noted in the Office Action, the present application claims priority back to U.S. Provisional Appl. No. 60/204,196. However, the analysis presented in the Office Action considered only the "direct" disclosure of the parent application, 09/853,823, and failed to consider the subject matter disclosed in the provisional application. The complete disclosure of the present application includes the subject matter that is directly disclosed in the non-provisional 09/853,823 as well as the provisional 60/204,196, both of which are incorporated

by reference into the present application. Thus, regardless of whether or not the parent application directly discloses enough supporting subject matter to obtain the benefit of priority to the parent application, the entire disclosure of the parent application, which includes the subject matter of the provisional application, discloses subject matter that supports the claim to priority in the present application. Therefore, the priority date for the present application is the date of the provisional, May 12, 2000, not December 28, 2001.

(B) Overview of General State of the Art and Various Embodiments of the Invention

It is thought that an overview of classic object-oriented programming and various embodiments of the invention, and combinations thereof, will assist in understanding how the disclosure of *Chau* differs from embodiments claimed in the application. This overview does not further limit any given claim beyond the language recited in the claim and is, accordingly, to be regarded in an illustrative rather than a restrictive sense.

It is well-known that with classic (i.e., conventional) object-oriented programming techniques, an instance of a class inherits attributes and methods from the class. However, as stated in the Background section, with classic object oriented programming, inheritance cannot be used (a) to provide different implementations for a particular method for instances from the same class or (b) to provide different attributes for instances from the same class. Thus, because classic inheritance does not allow for instances of the same class to behave differently or have different attributes, there is a need for per-instance methods and attributes.

The application describes mechanisms for enabling per-instance methods and attributes. A mechanism for implementing per-instance attributes involves using "categories," where an object inherits properties associated with a category class when the object is associated with a corresponding category object (page 8, lines 20-22). In other words, a category class provides a mechanism for categorizing an object (page 10, line 9). A purpose of the category mechanism

is to offer another level of flexibility over and above what multiple inheritance offers. As such, the category mechanism allows for associating attributes with one instance of a "main" class without having to associate the same attributes with any other instances of the same "main" class.

This category mechanism is different than conventional class inheritance, wherein all objects of the same class inherit the same properties (i.e., attributes and methods) from the class and/or subclass from which the objects are instantiated. Thus, with conventional class inheritance, all objects of the same class are identical in their properties, with the exception of particular values for attributes and/or arguments for methods that may vary from object to object.

(C) How *Chau* Differs From the Claims

Even though the application makes is clear that embodiments of the invention are in the context of objection oriented programming, Claim 1 is amended to further clarify this context. Claim 1 is amended to recite that the data item is an instance of a first class <u>and inherits attributes and methods from the first class</u>. As discussed above, this is a well-known relationship, in object-oriented programming, between a class and the instances (or objects) of the class. Hence, one skilled in the art of object-oriented programming is certainly familiar with this most fundamental relationship, inheritance. Furthermore, this fundamental relationship is discussed throughout the application. Therefore, no new matter is introduced in the application by way of this amendment to Claim 1 and <u>no new search</u> is necessary in response to this amendment.

Claim 1 recites, *inter alia*, associating a data item, which is an instance of a first class and inherits attributes and methods from the first class, with a category object that is an instance of a category class. The category class is a different class from the first class. Significantly, the

data item is associated with the category object <u>without associating the category object with all</u> <u>other instances of the first class</u>. Consequently, the data item is now associated with a data structure that includes storage for <u>values for one or more attributes of the category class</u>. A practical result of this is that one object of the first class can be associated with the category object and, therefore, inherit properties of the category class, whereas another object of the same first class is not associated with the category object and does not inherit the properties of the category class.

In the rejection of Claim 1, the Office Action relies on the following:

(1) an "order.xml" document as an instance of an "xml file type" for an alleged teaching of the data item being an instance of first class;

(2) a "LineItem.dtd" document type definition as an object of a "DTD file type" for an alleged teaching of the category object being an instance of a category class; and

(3) general storage of XML data in a relational database as a non-traditional data type or as traditional data in relational tables, for a teaching of "associating said data item with said category object without associating said category object with all other instances of said first class thereby causing said data item to be associated with a structure that includes storage for values for said one or more attributes."

One feature of Claim 1 that is missing from *Chau* is that an instance inherits attributes and methods from a class. The Office Action contends that an XML document is an instance of an XML file type. This is an inaccurate analogy. An "XML file type" does not define attributes and methods that XML documents inherit and, therefore, is not a "class" as used in object-oriented programming. An "XML file type" simply means that a file of that type is based on the Extensible Markup Language (i.e., XML) and is identified by its ".xml" file extension.

However, neither the fact that a document is of .xml file type, or that the document is written in XML, means that the document <u>inherits methods</u> and attributes from a class.

Perhaps a more accurate analogy to the object oriented programming instance-class relationship is the relationship between an XML document and a corresponding XML schema or DTD to which the XML document conforms. In other words, perhaps an XML document can be considered an instance of an XML schema or DTD, but that is not the position of the Office Action. The Office Action contends that an XML document is an instance of an XML file type. Even if the rejection of Claim 1 relied on the XML document-XML schema relationship, <u>XML schemas do not define *methods* that are inherited by documents that conform to the schema</u>.

The Office Action also contends that a DTD (LineItem.dtd) is an instance of a DTD file type. This too is an inaccurate analogy. A "DTD file type" does not define attributes and methods that DTDs inherit and, therefore, is not a "class" as used in object-oriented programming. A "DTD file type" simply means that a file of that type is a DTD. Even if the rejection of Claim 1 relied on the XML document-XML DTD relationship, <u>XML DTDs do not define *methods* that are inherited by documents that conform to the DTD</u>.

Claim 1 recites a programmatic mechanism for enabling per-instance attributes, through which an instance of a category class is associated with a particular instance of a "primary" class, but not all instances of the primary class. Hence, the particular instance can inherit attributes from the category class, in addition to the attributes and methods it inherits from its primary class. *Chau* does not disclose an instance of a class inheriting attributes from a class (i.e., a category class) that is not in the conventional lineage of the instance, nor does *Chau* disclose attributes from a class (i.e., the category class) that can be inherited by instances of different classes in different lineages. This category mechanism is different than conventional

class inheritance, wherein all objects of the same class inherit the same properties (i.e., attributes and methods) from the class and/or subclass from which the objects are instantiated. Thus, with conventional class inheritance, all objects of the same class are identical in their properties, with the exception of particular values for attributes and/or arguments for methods that may vary from object to object.

Based on the foregoing, *Chau* does not anticipate Claim 1. **Claims 2-13 and 16-23** depend, directly or indirectly, from Claim 1 and are patentable over the references of record for at least the same reasons as Claim 1. Furthermore, each of Claims 2-23 includes at least one other limitation that makes it further patentable over the references of record. However, due to the fundamental difference between Claim 1 and *Chau* discussed above, discussion of these additional differences is unnecessary and is foregone at this time. However, the rejection of the dependent claims is collectively traversed, and no statements of official notice or allegations of well-known features that may be present in the Office Action are stipulated to or admitted as prior art features, and the right to separately argue such features in the future is not disclaimed. Withdrawal of the rejection of Claims 1-13 and 16-23 under 35 U.S.C. §102(e) is requested.

Independent **Claim 24** recites similar limitations as Claim 1, specifically, regarding associating a data item, which is an instance of a class and <u>inherits attributes and methods from said class</u>, with two category objects thereby causing the data item to be associated with a structure that includes storage for values for one or more attributes of each of the two category classes from which the category objects are instantiated. As shown above, *Chau* is absent a teaching or suggestion of the data item being associated with a structure that includes storage for values for one or more attributes of a category class from which the attributes are inherited. Therefore, *Chau* is also absent a teaching or suggestion of the data item being associated with a

structure that includes storage for values for one or more attributes of each of multiple category classes (i.e., the data item inherits attributes from each of the first and second category classes).

Based on the foregoing, *Chau* does not anticipate Claim 24. **Claims 25-36 and 39-45** depend, directly or indirectly, from Claim 24 and are patentable over the references of record for at least the same reasons as Claim 24. Furthermore, each of Claims 25-45 includes at least one other limitation that makes it further patentable over the references of record. However, due to the fundamental difference between Claim 24 and *Chau* discussed above, discussion of these additional differences is unnecessary and is foregone at this time. However, the rejection of the dependent claims is collectively traversed, and no statements of official notice or allegations of well-known features that may be present in the Office Action are stipulated to or admitted as prior art features, and the right to separately argue such features in the future is not disclaimed. Withdrawal of the rejection of Claims 24-36 and 39-45 under 35 U.S.C. §102(e) is requested.

Rejection under 35 U.S.C. §103(a)

Claims 14, 15, 37 and 38 were rejected under 35 U.S.C. §103(a) as allegedly unpatentable over *Chau* in view of *Ng* in view of *Bowen*. This rejection is traversed.

Claims 14 and 15 depend from Claim 1 and Claims 37 and 38 depend from Claim 24. As shown above, *Chau* is deficient in its teachings in regards to Claims 1 and 24, the independent claims from which Claims 14, 15, 37 and 38 respectively depend. Furthermore, neither *Ng* nor *Bowen* cure the deficiencies in the teachings of *Chau*. Therefore, Claims 14, 15, 37 and 38 are also patentable over the cited references of record. Withdrawal of the rejection of Claims 14, 15, 37 and 38 under 35 U.S.C. §103(a) is requested.

THE CLAIM OBJECTIONS

Claims 3, 26, 32-37 and 39-43 were objected to because of informalities. Claims 3, 26, 32-37 and 39-43 are amended herein to correct the clerical errors that led to the objections of these claims. The objections to these claims are now moot. Based on the purpose of the claim amendments being to correct clerical errors, these amendments do not introduce new subject matter in the application, nor do they require a new search.

CONCLUSION

For the reasons set forth above, it is respectfully submitted that all of the pending claims (1-45) are in condition for allowance. Therefore, the issuance of a formal Notice of Allowance is believed next in order, and that action is most earnestly solicited.

The Examiner is respectfully requested to contact the undersigned by telephone if it is believed that such contact would further the examination of the present application.

Please charge any shortages or credit any overages to Deposit Account No. 50-1302.

Respectfully submitted,

HICKMAN PALERMO TRUONG & BECKER LLP

Date: 9/30/05

John D. Henkhaus
Reg. No. 42,656

2055 Gateway Place, Suite 550
San Jose, CA 95110-1089
(408) 414-1080
Facsimile: (408) 414-1076